BUNDESREPUBLIK DEUTSCHLAND

Prioritätsbescheinigung über die Einreichung einer Patentanmeldung

Aktenzeichen:	103 18 031.1
Anmeldetag:	19. April 2003
Anmelder/Inhaber:	DaimlerChrysler AG, 70567 Stuttgart/DE
Bezeichnung:	Verfahren zur Sicherstellung der Integrität und Authentizität von Flashware für Steuergeräte
IPC:	G 06 F 13/00

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.

München, den 18. März 2004
Deutsches Patent- und Markenamt
Der Präsident
Im Auftrag

Stanschus

A 9161
08/00
EDV-L

DaimlerChrysler AG

Eschbach
14.04.2003

Verfahren zur Sicherstellung der Integrität und Authentizität von Flashware für Steuergeräte

Die Erfindung betrifft ein Sicherheitskonzept für den Down loadprozess einer Software in einem Steuergerät.

Mit dem zunehmenden Anteil der Elektronik und der Kommunikationsmöglichkeiten im und mit einem Fahrzeug wachsen auch die Anforderungen, welche an die Sicherheit gestellt werden müssen. In den verschiedenen Bereichen der Technik werden heutzutage Mikrocontroller zur Steuerung eingesetzt. Diese Steuergeräte sind heutzutage oft über ein Bussystem miteinander verbunden und es gibt meist Möglichkeiten von außen auf diesen Bus zuzugreifen und mit den einzelnen Steuergeräten zu kommunizieren. Die Funktionsweise der Steuergeräte wird hierbei durch Anwendungsprogramme bestimmt. Diese Anwendungsprogramme sind bisher meist in einem nicht programmierbaren Speicher, bevorzugterweise im Steuergerät abgelegt. Dadurch ist eine Manipulation der Software nicht ohne weiteres zu realisieren. Beispielsweise kann der komplette Austausch eines Speicherbausteins gegen einen anderen Speicherbaustein erkannt und entsprechend darauf reagiert werden. Durch den zukünftigen Einsatz von programmierbaren, insbesondere sogenannten flashprogrammierbaren Steuergeräten im Fahrzeug, wird die Gefahr jedoch größer, dass unbefugte Manipulationen an den Anwendungsprogrammen und somit an der Arbeitsweise der Steuergeräte durchgeführt werden. Es müssen deshalb Maßnahmen getroffen werden, die ein unbefugtes Überschreiben von Anwendungsprogrammen in den Steuergeräten verhindern.

Einen typischen Downloadprozess eines Anwendungsprogramms, einer sogenannten Flashware, offenbart die Patentschrift DE 195 06 957 C2. Bei diesem System wird ein Anwendungsprogramm, eine sogenannte Flashware, in einen nichtflüchtigen, elektrisch lösch- und programmierbaren Speicher, in der Fachwelt als Flash-EPROM-Speicher bekannt oder kurz als Flash bekannt, abgelegt. Hierzu ist im elektrisch lösch- und programmierbaren Schreiblesespeicher (Flash) eine Initialisierungsroutine im Boot-Bereich hinterlegt. Mit Hilfe dieser Initialisierungsroutine werden die Anwenderprogramme bei der Inbetriebnahme des Mikroprozessorsystems geladen und gestartet. Um ein bestehendes Anwendungsprogramm durch ein neues ersetzen zu können, enthält die Initialisierungsroutine zusätzlich eine sogenannte Nachladeroutine. Diese Nachladeroutine wird über eine Systemschnittstelle mittels eines besonderen Befehles aktiviert. Nach Aktivierung speichert die Nachladeroutine das neue Anwendungsprogramm zunächst in einem Zwischenspeicher ab. Mittels eines zyklischen Blocksicherungsverfahrens wird überprüft, ob die Abspeicherung des neuen Anwendungsprogramms fehlerhaft war oder nicht. Wurde das neue Abwendungsprogramm korrekt übertragen und zwischengespeichert, wird das Löschen des auszutauschenden Anwenderprogramms eingeleitet und durchgeführt. Hierzu wird das alte Anwendungsprogramm in dem lösch- und programmierbaren Schreiblesespeicher (Flash) mit dem neuen Anwendungsprogramm überschrieben. Auch dieser Programmier- bzw. Kopiervorgang in den Flash kann mittels eines zyklischen Blocksicherungsverfahrens überprüft werden. Die Überprüfung mittels zyklischen Blocksicherungsverfahrens erlaubt lediglich eine Überprüfung inwieweit das Programm korrekt kopiert wurde. Eine Überprüfung auf Datenintegrität und Authentizität ist mit zyklischen Blocksicherungsverfahren nicht möglich. Ein nicht autorisiertes Programm bzw. eine nicht autorisierte Flashware kann mit zyklischen Blocksicherungsverfahren nicht erkannt werden.

Andererseits kennt man aus dem Bereich des Internets, besonders für Homebanking- und Homeshopping-Anwendungen, Ver-

schlüsselungsverfahren und digitale Signaturverfahren. Die Basis aller heute verbreiteten Verschlüsselungsverfahren ist die sogenannte Public-Key-Verschlüsselung. Diese Verschlüsselungsalgorithmen arbeiten mit einem geheimen und einem öffentlichen Schlüssel, bei dem der öffentliche Schlüssel öffentlich bekannt sein darf, wogegen der geheime Schlüssel nur einer autorisierten Stelle, beispielsweise einem Trust-Center bekannt sein darf. Solche kryptographischen Algorithmen sind z. B. Rivest, Shamir und Adleman (RSA-Algorithmus), Data Encryption Algorithmus (DEA-Algorithmus) oder dgl. Mit dem geheimen oder öffentlichen Schlüssel lässt sich - analog zur handschriftlichen Unterschrift - eine digitale Signatur zu einem elektronischen Dokument erzeugen. Nur der Besitzer des geheimen bzw. öffentlichen Schlüssels kann eine gültige Signatur erstellen. Die Echtheit des Dokuments kann dann über die Verifikation der Unterschrift mittels des zugehörigen öffentlichen bzw. geheimen Schlüssel geprüft werden. Der geheime Schlüssel wird manchmal auch als privater Schlüssel bezeichnet.

Als Signaturverfahren ist die elektronische Unterschrift bekannt geworden. Bei der elektronischen Unterschrift geht es darum sicherzustellen, dass eine Nachricht mit Sicherheit von einem bestimmten Absender kommt und dass diese Nachricht während der Übertragung nicht verfälscht wurde.

Hat der Sender erst einmal einen öffentlichen und einen privaten Schlüssel erzeugt, so ist folgendes Verfahren denkbar:

Der Sender der Informationen verschlüsselt mit seinem eigenen privaten Schlüssel eine Nachricht, die mit dem öffentlichen Schlüssel des Senders gelesen werden kann. Eine Nachricht, die mit öffentlichen Schlüssel gelesen werden kann, kann nur von dem Sender stammen, denn nur der Sender hat den passenden privaten Schlüssel. Hierbei gilt, dass man mit dem privaten Schlüssel nur verschlüsseln kann, während man mit dem öffentlichen Schlüssel nur entschlüsseln bzw. lesen kann. So ent-

stehen also Nachrichten, die nur von einer Person geschrieben, aber von allen, die den öffentlichen Schlüssel haben, gelesen werden können.

Die Verschlüsselung der gesamten Nachricht ist mit den vorgenannten Verschlüsselungsverfahren relativ rechenzeitaufwendig und für den Zweck nur die Authentizität des Autors festzustellen nicht notwendig. Daher wird in der Praxis ein etwas anderes Verfahren verwendet:

- Der Sender berechnet eine Art Zusammenfassung oder Quersumme der Nachricht, den sogenannten Hash-Code. Dabei ist die Berechnungsvorschrift so beschaffen, dass es nahezu unmöglich ist, die Nachricht zu verändern, ohne gleichzeitig den Hash-Code zu ändern.

- Der Sender verschlüsselt dann den Hash-Code mit seinem privaten Schlüssel. Das ist die elektronische Unterschrift. Die Unterschrift ist also für jede Nachricht anders, nur die Länge der Unterschrift ist immer gleich, unabhängig von der Länge der Nachricht. Dies ist Eigenschaft der Hash-Codes, die immer die gleiche Länge haben.

- Versandt wird dann die Nachricht mit der Unterschrift.

- Der Empfänger entschlüsselt die Unterschrift mit dem öffentlichen Schlüssel des Senders und erhält den vom Sender ermittelten Hash-Code.

- Nun kann der Empfänger selbst den Hash-Code der Originalnachricht ermitteln und mit dem Hash-Code, der vom Sender mitgeschickt wurde, vergleichen. Stimmen beide Hash-Codes überein, ist sichergestellt, dass die Nachricht wirklich von dem einen Sender stammt und dass sie auf dem Übertragungsweg nicht verfälscht wurde.

Der vorgenannte Signaturmechanismus basiert hierbei für die Dechiffrierung auf dem Public-Key-Verfahren RSA und für die Berechnung des Hash-Codes auf der Hash-Funktion RIPEMD-160.

Durch Kombination von Verschlüsselung und elektronischer Unterschrift können schließlich Nachrichten versandt werden, die vor Verfälschung sicher und eindeutig einem Absender zuzuordnen sind.

Basierend auf den vorgenannten Verschlüsselungsverfahren und Signaturverfahren hat man in der deutschen Patentanmeldung DE 100 08 974 A1 ein Signaturverfahren zur Sicherstellung der Datenintegrität einer Software für ein Steuergerät in einem Kraftfahrzeug vorgeschlagen. Bei diesem Verfahren wird der öffentliche Schlüssel in einem Speicherbereich des Steuergerätes hinterlegt. Die einzuspielende Software, die sogenannte Flashware, wird mit dem zweiten geheimen Schlüssel signiert. Zum Einspielen der signierten Software wird diese Flashware zunächst in einem Speicher des Steuergerätes abgelegt. Mit dem im Steuergerät selbst hinterlegten öffentlichen Schlüssel wird die Signatur der Flashware überprüft. Wenn die Überprüfung der elektronischen Signatur mit positivem Ergebnis verläuft, wird die zwischengespeicherte Flashware in einen elektrisch löschbaren und programmierbaren Speicher auf dem Steuergerät, den sogenannten Flash, eingelesen.

Zur Berechnung der Public-Key-Algorithmen sind nicht alle Steuergeräte in der Lage, da sie teilweise keine Gleitkommaarithmetik unterstützen oder nicht ausreichend Speicherplatz zur Verfügung stellen können. Um RSA sicher gestalten zu können, sollten als Schlüssellänge derzeit mindestens 1024 Byte gewählt werden. Die vorgenannten Signaturverfahren können deshalb in vielen, der heute in Fahrzeugen verwendeten Steuergeräten nicht eingesetzt werden.

Ausgehend von dem vorgenannten Stand der Technik ist es erfindungsgemäße Aufgabe, ein vereinfachtes Signaturverfahren

anzugeben, das auf möglichst allen Steuergeräten in heutigen Kraftfahrzeugen eingesetzt werden kann.

Die erfindungsgemäße Lösung dieser Aufgabe gelingt mit einem Verfahren mit den Merkmalen der unabhängigen Ansprüche. Weitere vorteilhafte Ausgestaltungen der Erfindung sind in den Unteransprüchen und in der Beschreibung der Ausführungsbeispiele enthalten.

Die Lösung gelingt mit einem vereinfachten symmetrischen, kryptographischen Verfahren. Grundlage dieses Verfahrens ist ein Authentifizierungscode. Dieser Authentifizierungscode wird in einem gesicherten Bereich, einem sogenannten Trust-Center, berechnet, indem das Anwendungsprogramm, die sogenannte Flashware, mit einem geheimen Datenstring konkateniert wird und von dem konkatenierten Anwendungsprogramm ein Hash-Wert berechnet wird. Dieser Hash-Wert wird hierbei sowohl über das Anwendungsprogramm als auch über den geheimen Datenstring berechnet. Dieser Hash-Wert ist der Authentifizierungscode für das zu prüfende Anwendungsprogramm. Die Überprüfung des Authentifizierungscodes erfolgt in dem Mikroprozessorsystem oder in dem Steuergerät, in dem das Anwendungsprogramm eingesetzt werden soll. Hierzu ist in dem Mikroprozessorsystem oder dem Steuergerät ein zweiter, gleicher, geheimer Datenstring abgelegt. In das Mikroprozessorsystem bzw. in das Steuergerät wird zunächst das unverschlüsselte Anwendungsprogramm und der Authentifizierungscode übertragen. Dann wird im Mikroprozessorsystem bzw. im Steuergerät das unverschlüsselte Anwendungsprogramm mit dem zweiten gleichen, geheimen Datenstring konkateniert. Von diesem konkatenierten Anwendungsprogramm wird im Mikroprozessorsystem bzw. im Steuergerät ein Hash-Wert berechnet. Stimmen berechneter Hash-Wert und übertragener Authentifizierungscode überein, so gilt das übertragene Anwendungsprogramm bzw. die übertragene Flashware als authentisch und darf im Flashspeicher abgelegt werden und im Steuergerät bzw. im Mikroprozessorsystem angewandt werden. In einer Weiterbildung der Erfindung wird das

Anwendungsprogramm mit dem geheimen Datenstring beidseitig sowohl am Programmanfang als auch am Programmende konkateniert. Die Hash-Wertberechnung erfolgt dann über das beidseitig konkatenierte Anwendungsprogramm. Zur Überprüfung des dermaßen gebildeten Authentifizierungscodes wird im Mikroprozessorsystem bzw. im Steuergerät das unverschlüsselt übertragene Anwendungsprogramm mit dem im Steuergerät abgelegten zweiten, geheimen Datenstring ebenfalls beidseitig konkateniert und über das beidseitig konkatenierte Anwendungsprogramm im Steuergerät bzw. im Mikroprozessorsystem ein Hash-Wert gebildet. Stimmt der im Steuergerät bzw. Mikroprozessorsystem berechnete Hash-Wert mit dem übertragenen Authentifizierungscode überein, so gilt das übertragene Anwendungsprogramm als authentisch. Die beidseitige Konkatenierung hat den Vorteil eines verbesserten Schutzes gegenüber unerlaubten Manipulationen der Anwendungssoftware.

Eine weitere Verbesserung gegenüber Manipulationen erhält man mit einer zweimaligen Berechnung eines Hash-Wertes. Bei dieser Ausführung der Erfindung wird das Anwendungsprogramm zunächst einseitig mit einem geheimen Datenstring konkateniert, dann wird von dem einseitig konkatenierten Anwendungsprogramm ein Hash-Wert berechnet. Die Konkatenierung kann hierbei am Programmanfang oder am Programmende sein. Dieser erste Hash-Wert HMAC1 wird wiederum einseitig mit dem geheimen Datenstring konkateniert. Die Konkatenierung kann hierbei auf jeder Seite des ersten Hashwerts erfolgen. In einem weiteren folgenden Schritt wird dann zur Berechnung eines Authentifizierungscodes schließlich ein zweiter Hash-Wert HMAC über das Gesamtgebilde aus Datenstring und erstem Hash-Wert HMAC1 berechnet. Zur Überprüfung des Authentifizierungscodes im Steuergerät bzw. im Mikroprozessorsystem müssen im Mikroprozessorsystem bzw. im Steuergerät die vorgenannten Berechnungsschritte in gleicher Reihenfolge wiederholt werden. Stimmen der berechnete Hash-Wert mit dem übertragenen Authentifizierungscode überein, so gilt die übertragene Anwendungssoftware als einwandfrei.

Für den Downloadprozess von der Flashware selbst, gibt es verschiedene Möglichkeiten der Übertragung. Flashware und Authentifizierungscode können zusammen auf dem gleichen Vertriebsweg übertragen werden oder der Authentifizierungscode kann von dem Anwendungsprogramm auf getrennte Vertriebswege übertragen werden. Bei der getrennten Übertragung ist es vorteilhaft, die Flashware bzw. das Anwendungsprogramm auf hardwaremäßigen Speichermedien zu vertreiben. Als bevorzugte Speichermedien kommen Compactdiscs, EPROMs oder Speicherkarten in Frage.

Bei erfolgreicher Authentifizierung des zu übertragenden Anwendungsprogramms in den Flashspeicher des Systems wird das neue Anwendungsprogramm vorzugsweise mit einer Kennung versehen, einem sogenannten Flag. Diese Kennung zeichnet das Anwendungsprogramm als das jeweils gültige Anwendungsprogramm aus.

Mit der Erfindung werden hauptsächlich folgende Vorteile erzielt:

Zur Berechnung der Public-Key-Algorithmen sind nicht alle Steuergeräte in der Lage, da sie teilweise keine Gleitkommaarithmetik unterstützen oder nicht ausreichend Speicherplatz zur Verfügung stellen können, um die erforderlichen Verschlüsselungsberechnungen durchführen zu können. Um die Public-Key-Algorithmen sicher zu gestalten, sollten als Schlüssellänge derzeit mindestens 1024 Byte gewählt werden. Da viele Steuergeräte in Kraftfahrzeugen lediglich einen Speicherbereich von 4 KByte haben, würde alleine schon der Schlüssel einen großen Teil des Speichers belegen. Die Erfindung kommt hier ohne Verschlüsselungsalgorithmen aus. Das einzige Berechnungsverfahren, das eingesetzt wird, ist die Hash-Wertberechnung. Mit Hilfe des erfindungsgemäßen symmetrischen, kryptographischen Verfahrens lassen sich auch diejenigen Steuergeräte mit einer Authentizitätsprüfung ausstatten, für die Public-Key-Verfahren nicht anwendbar sind.

Das erfindungsgemäße Verfahren ist ein sogenanntes Message Authentication Code-Verfahren, das auf der Berechnung eines Hash-Wertes basiert. Es ist damit kein Signaturverfahren. Ein Signaturverfahren erfordert, dass der Empfänger einer Nachricht nicht in der Lage ist, die mitgelieferte Signatur nachzubilden. Für die Anwendung in eingebetteten Systemen, wie z. B. Steuergeräten, ist ein Signaturverfahren nicht erforderlich, da das empfangende Steuergerät den Message Authentication Code für eine Nachricht nicht selbstständig bildet. Das Steuergerät prüft lediglich einen gegebenen geheimen Datenstring für eine gegebene Nachricht. Die Hash-Wertberechnung ist erforderlich, um den Übertragungsweg abzusichern. Erfindungsgemäß wird nämlich lediglich der Hash-Wert eines Message Authentication Codes übertragen und nicht der geheime Datenstring. Das erfindungsgemäß vorgeschlagene Hash-Wertverfahren ist wesentlich laufzeit- und speicherplatzeffizienter als es Chiffrier- und Dechiffrierverfahren, wie z. B. die Public-Key-Algorithmen, sein können.

In den Authentifizierungscode können Flashware-Metainformationen mit einbezogen werden. Flashware-Metainformationen sind z. B. der Speicherort der Flashware, die Identifikationsnummer der Flashware, die Identifikationsnummer des Steuergerätes oder die Fahrzeugidentifikationsnummer. Diese Flashware-Metainformation wird in den geheimen Datenstring integriert. Durch die Hash-Wertbildung über die Flashware und über den geheimen Datenstring ist damit sichergestellt, dass auch die Flashware-Metainformation auf dem Übertragungsweg gegenüber Manipulationen gesichert wird.

Kommt die gleiche Flashware auf mehreren Steuergeräten zum Einsatz, so kann durch Einbeziehung der Flashware-Metainformation in den Authentifizierungscode der Download-Vorgang der Flashware in die verschiedenen Steuergeräte mit diesem Authentifizierungscode selektiert werden. Da verschiedene Steuergeräte verschiedene Identifikationsnummern haben und auch die Speicherorte für die Flashware in den verschiedenen Steu-

ergeräten unterschiedlich ist, ergibt sich selbst bei gleicher Flashware nach dem erfindungsgemäßen Verfahren jeweils ein steuergerätespezifischer Authentifizierungscode.

Ausführungsbeispiele der Erfindung werden im Folgenden anhand der Figuren näher erläutert.

Es zeigen:

Fig. 1 schematisch einen Downloadprozess einer Flashware von einem Datenspeicher bis in das Steuergerät eines Kraftfahrzeugs;

Fig. 2 einen Downloadprozess für Flashware, bei dem die Flashware und der Authentifizierungscode auf getrennten Vertriebswegen in das Steuergerät eines Kraftfahrzeuges gelangen;

Fig. 3 ein Ablaufschema für die Berechnung eines Authentifizierungscodes für den Downloadprozess nach Figur 1;

Fig. 4 ein aufwendigeres Ablaufschema für die Berechnung eines Authentifizierungscodes für den Downloadprozess nach Figur 1;

Fig. 5 ein Ablaufschema für die Berechnung eines Authentifizierungscodes für den Downloadprozess nach Figur 2;

Fig. 6 ein aufwendigeres Ablaufschema für die Berechnung eines Authentifizierungscodes für den Downloadprozess nach Figur 2;

Fig. 7 ein Blockdiagramm eines Mikroprozessorsystems oder eines Steuergerätes mit einem Flashspeicher, in den mit dem erfindungsgemäßen Verfahren ein Anwendungsprogramm heruntergeladen werden kann.

Figur 1 zeigt eine Möglichkeit eines Downloadprozesses, bei dem Erfindung eingesetzt wird. Nach Abschluss der Programmentwicklung werden die Anwendungsprogramme bzw. die Flashware in einem Datenspeicher 1 gesammelt. Auf gesichertem Wege werden die einzelnen Anwendungsprogramme bzw. Anwendungs-RAM-

Pakete 2 in ein sogenanntes Trust-Center 3 überspielt. Im Trust-Center selbst, werden die Anwendungsprogramme mit einem Authentifizierungscode gekennzeichnet. Die Abläufe im Trust-Center selbst, werden weiter unten im Zusammenhang mit den Figuren 3 bis 6 näher erläutert. Vom Trust-Center aus, wird die unverschlüsselte Flashware zusammen mit dem Authentifizierungscode HMAC an eine externe Systemschnittstelle 4 übergeben. Die Systemschnittstelle selbst, kann im einfachsten Fall aus einem Diagnoseanschluss im Kraftfahrzeug bestehen. In der Regel wird jedoch die Systemschnittstelle durch das Diagnosesystem in den Kraftfahrzeugwerkstätten gebildet werden. Für die Übertragung vom Trust-Center zur Systemschnittstelle können hierbei die üblichen Datenkommunikationswege benutzt werden, das sind insbesondere Festnetzverbindungen, Internetverbindungen und auch Mobilfunkverbindungen. Von der Systemschnittstelle wird der Downloadprozess des übertragenen Programmpaketes bzw. der übertragenen Flashware und des Authentifizierungscodes HMAC in ein Steuergerät eines Kraftfahrzeuges veranlasst. Hierzu sendet die Systemschnittstelle an das Steuergerät im Kraftfahrzeug ein spezielles Kommando, mit dem der Flashspeicher im Kraftfahrzeug für den Downloadprozess vorbereitet wird. Das Einprogrammieren des neuen Anwendungsprogramms in den Flashspeicher wird weiter unten im Zusammenhang mit Figur 7 näher erläutert. Im Steuergerät des Kraftfahrzeugs wird der übertragene Authentifizierungscode HMAC überprüft und bei erfolgreicher Überprüfung wird die mitübertragene Flashware in den Flashspeicher des Steuergerätes einprogrammiert. Die Überprüfung des Authentifizierungscodes im Steuergerät des Kraftfahrzeuges erfolgt im Wesentlichen durch Wiederholen der Schritte, mit denen der Authentifizierungscode im Trust-Center erzeugt wurde. Nähere Erläuterungen zur Überprüfung des Authentifizierungscodes finden sich weiter unten in den Figurenbeschreibungen zu den Figuren 3 bis 6.

Figur 2 zeigt eine andere Möglichkeit eines erfindungsgemäßen Downloadprozesses. Auch bei diesem Ausführungsbeispiel werden die Anwendungsprogramme in einem Datenspeicher 1 gesammelt. Sodann werden die Anwendungsprogramme, die sogenannte Flashware, als Programmpakete 2 an ein Trust-Center übergeben. In dem Trust-Center 3 wird für die Flashware ein Authentifizierungscode erzeugt. Die Berechnung des Authentifizierungscodes wird weiter unten im Zusammenhang mit den Figuren 5 und 6 näher erläutert. Im Unterschied zu dem Downloadprozess nach Figur 1 wird bei dem hier beschriebenen Downloadprozess lediglich ein Authentifizierungscode HMAC vom Trust-Center an die Systemschnittstelle 4 übertragen. Das Anwendungsprogramm selbst, die sogenannte Flashware, wird auf einem getrennten Vertriebsweg übermittelt. Vorzugsweise wird die Flashware auf Compactdiscs, Speicherkarten, EPROMs oder anderen hardwaremäßigen Speichermitteln 6 festgehalten und mit einem geeigneten Lesegerät 7 in das Steuergerät 5 des Kraftfahrzeuges übertragen. Insbesondere bei Compactdiscs kann das geeignete Lesegerät 7 im Kraftfahrzeug durch ein Infotainmentsystem, wie es heute in Fahrzeugen eingesetzt wird, insbesondere durch ein CD-ROM-Laufwerk oder ein DVD-Laufwerk gebildet sein. Auch bei dem Ausführungsbeispiel nach Figur 2 wird im Fahrzeug der Downloadprozess durch ein spezielles Kommando von der Systemschnittstelle 4 eingeleitet. Hierzu hat die Systemschnittstelle 4 Zugriff auf die Datenbusse des Bordnetzes im Kraftfahrzeug. Mit einem Softwarekommando von der Systemschnittstelle wird das Einlesen der Flashware vom Lesegerät 7 in das Steuergerät 5 gestartet. Zugleich wird mit dem Softwarekommando der Flashspeicher des Steuergerätes 5 zur Übernahme der Flashware vorbereitet. Die Überprüfung des Authentifizierungscodes HMAC im Steuergerät wird weiter unten in den Figurenbeschreibungen zu Figur 5 und 6 näher erläutert. Im Prinzip müssen zur Überprüfung des Authentifizierungscodes die

Berechnungsschritte, die zur Erstellung des Authentifizierungscodes notwendig waren, in der gleichen Reihenfolge wie im Trust-Center im Steuergerät wiederholt werden. Auch bei diesem Ausführungsbeispiel kann die Systemschnittstelle im einfachsten Fall durch einen Diagnoseanschluss im Kraftfahrzeug gebildet sein. Bevorzugterweise ist jedoch die Systemschnittstelle das Diagnosesystem in der Kraftfahrzeugwerkstatt.

Die zuvor beschriebene Überprüfung des Authentifizierungscodes gilt unabhängig von der Wahl des Übertragungsweges für die Flashware. Der Authentifizierungsablauf ist beim Herunterladen der Flashware von CD-Rom oder DVD, der gleiche wie beim direkten Herunterladen der Flashware von einem Zentralsystem mittels drahtloser oder drahtgebundener Datenübertragung.

Allen Ausführungsbeispielen der Erfindung gemeinsam, ist die Berechnung eines Hash-Wertes. Mit Hilfe der Hash-Funktion, bekannt unter der Bezeichnung RIPEMD-160-Algorithmus, kann zu beliebig langen Daten ein Prüfwert, ein sogenannter Abdruck, fester Länge erzeugt werden. Dieser Abdruck wird als Hash-Wert bezeichnet. Hash-Funktion und Hash-Wert erfüllen dabei die folgenden Eigenschaften:

- Der Hash-Wert ist leicht zu berechnen.

- Es ist praktisch nicht möglich, bei gegebenem Hash-Wert einen Datensatz zu erzeugen, der diesen Hash-Wert liefert (Einwegfunktion). Zudem ist es schwer, eine Kollision, d. h. zwei Datensätze mit dem gleichen Hash-Wert zu finden (Kollisionsresistenz).

- Die Hash-Funktion kann nur für Daten bzw. Datensätze, deren Bitlänge maximal $2^{64} - 1$ ist, angewandt werden. Bei kürzeren Datensätzen werden die Datensätze mit Nullen aufgefüllt, bis die Länge de aufgefüllten Datensatzes ein ganzzahliges Vielfaches von 512 Bit hat. Der aufgefüllte Datensatz wird dann in mindestens 512 Bit-lange Blöcke aufgeteilt. Die Anwendung der Hash-Funktion auf die 512 Bit langen Blöcke ergibt schließlich einen 160 Bit langen Hash-Wert. Die Hash-Funktion kann hierbei auf beliebige Datensätze angewandt werden, insbesondere auch auf Flashware.

Figur 3 zeigt ein Ablaufschema zur Berechnung eines Authentifizierungscodes innerhalb eines gesicherten Bereiches 3, der im Folgenden als Trust-Center bezeichnet wird. Im Trust-Center werden in einem gesonderten, gesicherten Bereich, vorzugsweise einem gesonderten, gesicherten Datenspeicher 8, geheime Kennungen bzw. Identifizierungen in Form von Datenstrings in digitaler Form verwaltet. Die Flashware, für die ein Authentifizierungscode berechnet werden soll, wird zunächst an beiden Enden des Anwendungsprogramms mit einem Datenstring konkateniert. Das heißt, den digitalen Datensatz des Anwendungsprogramms wird am Anfang und am Ende ein geheimer Datenstring aus dem Speicher 8 des Trust-Centers angehängt. Im nächsten Schritt wird für die beidseitig konkatenierte Flashware ein Hash-Wert berechnet. Dieser Hash-Wert beinhaltet nun sämtliche Informationen über die Flashware sowie über den geheimen Datenstring. Durch die zuvor erläuterten Eigenschaften der Hash-Funktion ist dieser Hash-Wert als Authentifizierungscode HMAC für die Authentizität und die Datenintegrität der Flashware geeignet. In dem nächsten Schritt wird der Authentifizierungscode HMAC dem unverschlüsselten Anwendungsprogramm, der sogenannten Flashware, hinzugefügt

und vom Trust-Center an die Systemschnittstelle zur weiteren Übertragung in das Kraftfahrzeug übermittelt.

Figur 4 zeigt einen aufwendigeren Ablauf zur Berechnung eines Authentifizierungscodes in einem Trust-Center. Bei diesem Ausführungsbeispiel wird die Flashware zunächst einseitig mit einem geheimen Datenstring konkateniert. Die Konkatenierung kann hierbei sowohl am Anfang oder auch am Ende des Datensatzes der Flashware erfolgen. Über die einseitig konkatenierte Flashware wird eine erste Hash-Wertberechnung durchgeführt. Man erhält einen ersten Hash-Wert HMAC1. Dieser erste Hashwert HMAC1 wird wiederum einseitig mit einem geheimen Datenstring aus dem Speicher 8 konkateniert. Auch hier kann die Konkatenierung am Anfang oder am Ende des ersten Hashwerts erfolgen. In einem weiteren Schritt wird über das Gesamtgebilde aus geheimen Datenstring und erstem Hashwert eine zweite Hash-Wertberechnung durchgeführt. Das Ergebnis dieser letzten Hash-Wertberechnung ergibt den Authentifizierungscode HMAC. Sodann wird unverschlüsselte Original-Flashware zu dem Authentifizierungscode hinzugefügt und an die Systemschnittstelle übertragen. Das Ausführungsbeispiel der Figur 4 ist geeignet für einen Downloadprozess nach Figur 1.

Figur 5 zeigt ein Ablaufschema zur Berechnung eines Authentifizierungscodes innerhalb eines Trust-Centers zur Verwendung in dem Downloadprozess nach Figur 2. Im Trust-Center wird die unverschlüsselte Original-Flashware beidseitig mit einem geheimen Datenstring konkateniert. Im nächsten Schritt wird für die beidseitig konkatenierte Flashware eine Hash-Wertberechnung durchgeführt. Das Ergebnis dieser Hash-Wertberechnung ist der Authentifizierungscode HMAC. Im Unterschied zu dem Ausführungsbeispiel der Figur 3 wird bei dem Ausführungsbeispiel der Figur 5 lediglich der Authentifizierungscode an die Systemschnittstelle übertragen. Der Vertrieb

der Original- und unverschlüsselten Flashware erfolgt hierbei auf getrennten Vertriebswegen. Die Flashware wird hierbei vorzugsweise auf hardwaremäßigen Speicherelementen übermittelt und in das Kraftfahrzeug eingelesen (näheres hierzu siehe Figur 2).

Figur 6 zeigt ein weiteres Ausführungsbeispiel einer aufwendigeren Berechnung eines Authentifizierungscodes, wie er im Zusammenhang mit dem Downloadprozess nach Figur 2 Verwendung findet. Bei diesem Ausführungsbeispiel wird in dem Trust-Center die unverschlüsselte Flashware zunächst mit einem geheimen Datenstring einseitig konkateniert. Die Konkatenierung kann hierbei sowohl am Anfang als auch am Ende des Datensatzes der Flashware erfolgen. Über die einseitig konkatenierte Flashware wird eine Hash-Wertberechnung durchgeführt. Das Ergebnis ist ein erster Hash-Wert HMAC1. Dieser erste Hashwert HMAC1 wird wiederum einseitig mit einem geheimen Datenstring aus dem Speicher 8 konkateniert. Auch hier kann die Konkatenierung am Anfang oder am Ende des ersten Hashwerts erfolgen. In einem weiteren Schritt wird über das Gesamtgebilde aus geheimen Datenstring und erstem Hashwert eine zweite Hash-Wertberechnung durchgeführt. Das Ergebnis dieser letzten Hash-Wertberechnung ergibt den Authentifizierungscode HMAC. Dieser Authentifizierungscode wird an die Systemschnittstelle übermittelt. Im Unterschied zu dem Ausführungsbeispiel der Figur 4 wird bei dem Ausführungsbeispiel der Figur 6 lediglich der Authentifizierungscode an die Systemschnittstelle übermittelt. Die unverschlüsselte Originalsoftware wird hierbei analog zur Figur 2 über Speichermedien, vorzugsweise Compactdiscs, in das Kraftfahrzeug eingelesen.

Anhand von Figur 7 wird im Folgenden auf den Flashprozess im Steuergerät bzw. im Mikroprozessorsystem des Kraftfahrzeuges näher eingegangen. Ein typisches Steuergerät, auch als Elect-

ronic Control Unit ECU bezeichnet, enthält eine Recheneinheit, einen sogenannten Mikroprozessor CPU, der über einen Prozessorbus PBUS mit verschiedenen Speichern bzw. Speichersektoren verbunden ist. Über ein Interface kann das Steuergerät entweder von außen angesprochen werden oder mit anderen, an das Interface angeschlossenen Einheiten kommunizieren. Der Speicher des Steuergerätes besteht aus einem Boot-Sektor, einem Flashspeicher und einem Arbeitsspeicher RAM. Der Flashspeicher Flash ist ein elektrisch lösch- und programmierbarer Speicher, beispielsweise ein EEPROM. Im Boot-Sektor ist das Betriebssystem des Mikroprozessors, ein sogenannter Flash Boot Loader, sowie der RIPEMD-160-Algorithmus für die Hash-Funktion abgelegt. In einem, von äußeren Zugriffen besonders geschützten Speicher bzw. Speicherbereich ist in dem Steuergerät ein geheimer Datenstring abgelegt. Dieser besonders geschützte Datenbereich 9 kann auch im Boot-Bereich angeordnet sein. Eine andere Möglichkeit ist die Ausbildung dieses besonders geschützten Datenspeichers 9 in Form einer nicht überschreibbaren und vor unbefugtem Auslesen geschützten Speicherkarte oder in Form eines sogenannten Kryptoprozessors, der seinen Inhalt bei dem Versuch eines unberechtigten Zugriffs löscht. Durch diese Maßnahmen bzw. durch diese Ausbildung des besonders geschützten Speicherbereiches 9 wird die Geheimhaltung des darin abgespeicherten Datenstrings gesichert. Welche Datenstrings in den besonders geschützten Datenbereich 9 einprogrammiert werden, muss mit den Datenstrings zur Berechnung der Authentifizierungscodes im Trust-Center koordiniert werden. Der Datenstring im Steuergerät muss mit dem Datenstring, der zur Grundlage der Berechnung des Authentifizierungscodes diente, übereinstimmen.

In dem Flash des Steuergerätes sind die Anwendungsprogramme hinterlegt, die als Flashware aktualisiert werden können. Eine Überschreibung, bereits hinterlegter Anwenderprogramme

durch neue Flashware, erfolgt grundsätzlich auf folgende Weise. Mit einem speziellen Softwarekommando, das von einer externen Systemschnittstelle über das Interface des Steuergerätes übertragen wird, wird das Steuergerät für einen Downloadprozess und für einen Flashprozess vorbereitet. Mit dem Softwarekommando wird der sogenannte Flash Boot Loader aktiviert. Der Flash Boot Loader ist im Wesentlichen eine Nachladeroutine, mit der Anwendungsprogramme in den Flashspeicher des Steuergerätes geschrieben werden. Beim Downloadprozess wird die neue Flashware und der übertragene Authentifizierungscode zunächst im Arbeitsspeicher des Steuergerätes zwischengespeichert. Dann wird, mit Hilfe der Nachladeroutine des Flash Boot Loaders, im Mikroprozessor des Steuergerätes die Überprüfung der zwischengespeicherten Flashware und des zwischengespeicherten Authentifizierungscodes auf Authentizität und Datenintegrität durchgeführt. Diese Überprüfung erfolgt derart, dass in dem Mikroprozessor mit der unverschlüsselten Software und dem im Steuergerät abgelegten geheimen Datenstring die gleichen Verfahrensschritte durchgeführt werden, die angewandt wurden, um den übertragenen Authentifizierungscode zu erzeugen. Es werden also im Mikroprozessor des Steuergerätes diejenigen Verfahrensschritte wiederholt, die im Trust-Center durchgeführt wurden, um den Authentifizierungscode zu erzeugen. Wurde z. B. der Authentifizierungscode nach dem Ausführungsbeispiel der Figur 3 erzeugt, so wird nun im Mikroprozessor des Steuergeräts die zwischengespeicherte Flashware beidseitig mit dem im Steuergerät abgelegten geheimen Datenstring konkateniert. Von der beidseitig konkatenierten Flashware wird mit dem RIPEMD-160-Algorithmus eine Hash-Wertberechnung durchgeführt. Das Ergebnis dieser Hash-Wertberechnung im Steuergerät wird mit dem übertragenen Identifizierungscode HMAC verglichen. Sind beide Hash-Werte identisch, so gilt die im Arbeitsspeicher zwischengespeicherte Flashware als authentisch und integer. Wurde der Authentifi-

zierungscode im Trust-Center nach einem der Ausführungsbeispiele entsprechend Figuren 3, 4, 5 oder 6 ermittelt, so müssen zur Überprüfung im Steuergerät bzw. im Mikroprozessor des Steuergeräts die Konkatenierungen der zwischengespeicherten Flashware mit dem im Steuergerät abgelegten geheimen Datenstring sowie die Hash-Wertberechnungen der konkatenierten Flashware in derjenigen Weise durchgeführt werden, wie sie jeweils im Trust-Center durchgeführt wurden, um den übermittelten Authentifizierungscode zu erzeugen. Ein Vergleich des Mikroprozessor des Steuergerätes ermittelten Hash-Werts mit dem übertragenen Authentifizierungscode gibt bei Übereinstimmung der beiden Werte jeweils eine Aussage zur Datenintegrität und Authentizität der übertragenen und im Arbeitsspeicher zwischengespeicherten Flashware. Stimmen beide Wert überein, gilt die Flashware jeweils als unbedenklich.

Nach erfolgreicher Überprüfung, der neu heruntergeladenen und zwischengespeicherten Flashware, schreibt der Flash Boot Loader die neue zwischengespeicherte Flashware in den Flashspeicher des Steuergerätes ein. Der Kopiervorgang von dem Arbeitsspeicher in den Flashspeicher kann hierbei zusätzlich mit einem zyklischen Blocksicherungsverfahren auf Vollständigkeit hin überprüft werden. War die Authentizitätsprüfung und der Kopiervorgang fehlerfrei, so wird für die nun im Flash befindliche Flashware ein sogenanntes Flag gesetzt. Dieses Flag kennzeichnet das nunmehr im Flash befindliche Anwendungsprogramm als das gültig zu verwendende Anwendungsprogramm. Das Flag kann hierbei wie in Figur 7 beispielhaft dargestellt z.B. im Flashspeicher selbst gesetzt werden, vorzugsweise ist der Flashspeicher als EEPROM ausgebildet. Das Steuergerät kann nun in die Anwendung gehen und wird dabei die mit einem gültigen Flag gekennzeichneten Anwendungsprogramme verwenden.

Die Aktivierung des Flash Boot Loaders erfolgt vorzugsweise durch das Diagnosesystem in einer Werkstatt. In diesem Fall bildet das Diagnosesystem der Werkstatt die Systemschnittstelle 4. Im Fall des Downloadprozesses nach Figur 1 können unverschlüsselte Flashware sowie Authentifizierungscode zusammen von der Systemschnittstelle über das Interface in den Arbeitsspeicher des Steuergerätes zwischengespeichert werden. Im Falle des Downloadprozesses nach Figur 2 wird der Authentifizierungscode über die Systemschnittstelle in den Arbeitsspeicher des Steuergerätes zwischengespeichert, während die unverschlüsselte Flashware über ein weiteres Lesegerät, vorzugsweise ein CD-ROM-Laufwerk bzw. ein Chipkartenlesegerät, in den Arbeitsspeicher des Steuergerätes zwischengespeichert wird. Bei dem Downloadprozess nach Figur 2 muss deshalb die Nachladeroutine des Flash Boot Loaders die benötigten Datensätze ggf. von unterschiedlichen EDV-Systemen herunterladen. In allen Fällen jedoch, erfolgt die Kommunikation im Kraftfahrzeug über die kraftfahrzeuginternen Datenbusse. Ein heutzutage weit verbreiteter Datenbus im Kraftfahrzeug ist der sogenannte CAN-Bus.

Nicht alle Steuergeräte in einem Kraftfahrzeug haben genügend Speicherplatz, um eine Zwischenspeicherung der Flashware durchführen zu können. Bei Steuergeräten, bei denen der vorhandene Speicherbereich nicht ausreicht, um die herunterzuladende Flashware zwischen zu speichern, wird der Downloadprozess deshalb wie folgt durchgeführt:

- Zunächst wird der vorhandene Flash Speicher gelöscht.

- Dann wird die neue Flashware heruntergeladen und einprogrammiert.

- Dann wird die heruntergeladenen Flashware verifiziert, das heißt auf Übertragungsfehler überprüft.

- Dann wird die Authentizitätsprüfung wie in den vorhergehenden Ausführungsbeispielen durchgeführt.

- Nach positiver Authentizitätsprüfung wird die heruntergeladene Flashware durch setzen eines Flags in Form eines Statusbits gekennzeichnet und aktiviert.

- Die folgenden Anwendungen greifen dann auf die neue Flashware zu.

Das direkte Herunterladen ohne Zwischenspeicherung der Flashware hat den zusätzlichen Vorteil einer „end-to-end"-Absicherung, da beim Schreibprozess während des Downloadprozesses auch Schreibfehler erkannt werden.

Bei allen Ausführungsbeispielen der Erfindung kann die Flashware um sogenannte Metainformationen ergänzt werden. Diese Flashware-Metainformation ist insbesondere eine Fahrzeugidentifizierungsnummer, eine Steuergerätesachnummer oder ein spezieller Speicherort für die Flashware. Durch Einbeziehung der Flashware-Metainformation lässt sich z. B. der Speicherort für die neu zu herunterladende Flashware auswählen. Dadurch dass die Flashware-Metainformation in die Berechnung des Authentifizierungscodes mit einbezogen ist, besteht auch ein Schutz gegenüber Manipulationen dieser Flashware-Metainformation.

DaimlerChrysler AG

Eschbach
14.04.2003

Patentansprüche

1. Verfahren zum Laden von zumindest einem aktuellen Anwendungsprogramm (Flashware), das in einem Programmspeicher (Flash) eines Mikroprozessorsystems (ECU) gespeichert wird, wobei an den Prozessorbus (PBUS) des Mikroprozessorsystems (ECU)
- mindestens ein Mikroprozessor (CPU),
- mindestens ein Programmspeicher mit einem Boot-Sektor, einem Flash Boot Loader, einem elektrisch löschbaren und programmierbaren Speicher (Flash) und einem Schreib-Lese-Speicher (RAM),
- sowie mindestens eine Systemschnittstelle (Diagnose-Interface, Bordnetz-Interface) angeschlossen sind,
und wobei
- für das Anwendungsprogramm (Flashware) ein Authentifizierungscode (HMAC) erstellt wird,
- der Authentifizierungscode (HMAC) und das aktuelle Anwendungsprogramm über die Systemschnittstelle eingelesen werden,
- und vor dem Aktivieren des eingelesenen aktuellen Anwendungsprogramms eine Überprüfung des an der Systemschnittstelle eingelesenen Authentifizierungscodes (HMAC) erfolgt,
d a d u r c h g e k e n n z e i c h n e t ,
dass der Authentifizierungscode (HMAC) in einem gesicherten Bereich (Trust-Center) berechnet wird, indem das Anwendungsprogramm (Flashware) mit einem geheimen Datenstring (STRING) konkateniert wird und von dem konkate-

nierten Anwendungsprogramm ein Hash-Wert berechnet wird, der als Authentifizierungscode (HMAC) an der Systemschnittstelle eingelesen wird, und dass im Mikroprozessorsystem ein zweiter, gleicher, geheimer Datenstring (STRING) abgelegt ist, mit dem das eingelesene Anwendungsprogramm (Flashware) im Mikroprozessorsystem konkateniert wird und von dem eingelesenen, konkatenierten Anwendungsprogramm im Mikroprozessor (CPU) ein Hash-Wert berechnet wird und mit dem übertragenen Authentifizierungscode (HMAC) verglichen wird.

2. Verfahren nach Anspruch 1,
d a d u r c h g e k e n n z e i c h n e t ,
dass das Anwendungsprogramm sowohl im gesicherten Bereich (Trust-Center) als auch bei der Authentizitätsprüfung im Mikroprozessor mit dem geheimen Datenstring am Programmanfang und am Programmende konkateniert und von dem beidseitig konkatenierten Anwendungsprogramm ein Hash-Wert berechnet wird, der als Authentifizierungscode (HMAC) an der Systemschnittstelle eingelesen wird.

3. Verfahren nach Anspruch 1,
d a d u r c h g e k e n n z e i c h n e t ,
- dass das Anwendungsprogramm zunächst entweder am Programmanfang oder am Programmende mit dem geheimen Datenstring (STRING) konkateniert wird,
- dass in einem folgenden Schritt von dem einseitig konkatenierten Anwendungsprogramm im gesicherten Bereich (Trust-Center) ein erster Hash-Wert (HMAC1) berechnet wird,
- dass in einem weiteren folgenden Schritt der erste Hash-Wert (HMAC1) einseitig mit einem geheimen Datenstring (STRING) konkateniert wird,
- dass in einem weiteren folgenden Schritt von dem Gesamtgebilde aus erstem Hashwert (HMAC1) und geheimem Datenstring (STRING) ein zweiter Hash-Wert (HMAC) berechnet wird, der als Authentifizierungscode (HMAC) an der Sys-

temschnittstelle eingelesen wird,
- und dass im Mikroprozessorsystem ein zweiter, gleicher, geheimer Datenstring (STRING) abgelegt ist, mit dem im Mikroprozessorsystem die im gesicherten Bereich (Trust-Center) durchgeführten Schritte mit dem ursprünglichen Anwendungsprogramm in gleicher Reihenfolge wiederholt werden,
- und der im Mikroprozessor berechnete Hash-Wert mit dem an der Systemschnittstelle eingelesenen Hash-Wert (HMAC) verglichen werden.

4. Verfahren nach einem der Ansprüche 1 bis 3,
d a d u r c h g e k e n n z e i c h n e t ,
dass der Authentifizierungscode (HMAC) zusammen mit dem Anwendungsprogramm (Flashware) übermittelt wird.

5. Verfahren nach einem der Ansprüche 1 bis 3,
d a d u r c h g e k e n n z e i c h n e t ,
dass der Authentifizierungscode (HMAC) getrennt von dem Anwendungsprogramm (Flashware) übermittelt wird.

6. Verfahren nach Anspruch 5,
d a d u r c h g e k e n n z e i c h n e t ,
dass das Anwendungsprogramm (Flashware) auf einem Speichermedium zwischengespeichert und mittels des Speichermediums vertrieben wird und der Authentifizierungscode (HMAC) mittels Datenübertragung vom gesicherten Bereich (Trust-Center) an die Systemschnittstelle übertragen wird.

7. Verfahren nach Anspruch 4,
d a d u r c h g e k e n n z e i c h n e t ,
dass das Anwendungsprogramm (Flashware) und der Authentifizierungscode (HMAC) mittels Datenübertragung vom gesicherten Bereich (Trust-Center) an die Systemschnittstelle übertragen werden.

8. Verfahren nach einem der Ansprüche 1 bis 7,
d a d u r c h g e k e n n z e i c h n e t ,
dass der Authentifizierungscode über die Diagnoseschnittstelle (Diagnose Interface) in ein Steuergerät (ECU) eines Kraftfahrzeuges eingelesen wird.

9. Verfahren nach einem der Ansprüche 1 bis 8,
d a d u r c h g e k e n n z e i c h n e t ,
dass wenn ein eingelesener Authentifizierungscode (HMAC) und im Mikroprozessor berechneter Hash-Wert übereinstimmen, das zugehörige Anwendungsprogramm (Flashware) mit einer Kennung (Flag) als gültiges Anwendungsprogramm versehen wird.

10. Verfahren nach einem der Ansprüche 1 bis 9,
d a d u r c h g e k e n n z e i c h n e t ,
dass in den Authentifizierungscode (HMAC) Flashware-Metainformation mit einbezogen wird.

11. Verfahren nach Anspruch 10,
d a d u r c h g e k e n n z e i c h n e t ,
dass mit dem Authentifizierungscode (HMAC) der Downloadprozess des Anwendungsprogramms auf verschiedene Steuergeräte selektiert wird.

12. Verfahren zur Sicherstellung der Authentizität von Flashware für ein Steuergerät (ECU) eines Kraftfahrzeugs, indem in einem Programmspeicher(Flash) ein Anwendungsprogramm gespeichert ist,
d a d u r c h g e k e n n z e i c h n e t ,
dass in einem gesicherten Bereich (Trust-Center) ein Authentifizierungscode (HMAC) berechnet wird, in dem das Anwendungsprogramm (Flashware) mit einem geheimen Datenstring (STRING) konkateniert wird und von dem konkatenierten Anwendungsprogramm ein Hash-Wert berechnet wird, der als Authentifizierungscode (HMAC) in das Steuergerät (ECU) eingelesen wird, und dass im Steuergerät (ECU) ein

zweiter, gleicher, geheimer Datenstring (STRING) abgelegt ist, mit dem das eingelesene Anwendungsprogramm (Flashware) im Steuergerät konkateniert wird und von dem eingelesenen, konkatenierten Anwendungsprogramm im Steuergerät (ECU) ein Hash-Wert berechnet wird und mit dem übertragenen Authentifizierungscode (HMAC) verglichen wird.

13. Verfahren nach Anspruch 12,
dadurch gekennzeichnet,
dass das Anwendungsprogramm sowohl im gesicherten Bereich (Trust-Center) als auch bei der Authentizitätsprüfung im Steuergerät (ECU) mit dem geheimen Datenstring am Programmanfang und am Programmende konkateniert und von dem beidseitig konkatenierten Anwendungsprogramm ein Hash-Wert berechnet wird, der als Authentifizierungscode (HMAC) an der Systemschnittstelle eingelesen wird.

14. Verfahren nach Anspruch 12,
dadurch gekennzeichnet,
- dass das Anwendungsprogramm zunächst entweder am Programmanfang oder am Programmende mit dem geheimen Datenstring (STRING) konkateniert wird,
- dass in einem folgenden Schritt von dem einseitig konkatenierten Anwendungsprogramm im gesicherten Bereich (Trust-Center) ein erster Hash-Wert (HMAC1) berechnet wird,
- dass in einem weiteren folgenden Schritt der erste Hash-Wert (HMAC1) einseitig mit einem geheimen Datenstring (STRING) konkateniert wird,
- dass in einem weiteren folgenden Schritt von dem Gesamtgebilde aus erstem Hashwert (HMAC1) und geheimem Datenstring (STRING) ein zweiter Hash-Wert (HMAC) berechnet wird, der als Authentifizierungscode (HMAC) an der Systemschnittstelle eingelesen wird,
- und dass im Steuergerät (ECU) ein zweiter, gleicher, geheimer Datenstring (STRING) abgelegt ist, mit dem im Steuergerät (ECU) die im gesicherten Bereich (Trust-

Center) durchgeführten Schritte mit dem ursprünglichen Anwendungsprogramm in gleicher Reihenfolge wiederholt werden,
- und der im Steuergerät berechnete Hash-Wert mit dem an der Systemschnittstelle eingelesenen Hash-Wert (HMAC) verglichen werden.

15. Verfahren nach einem der Ansprüche 12 bis 14, dadurch gekennzeichnet, dass der Authentifizierungscode (HMAC) zusammen mit dem Anwendungsprogramm (Flashware) übermittelt wird.

16. Verfahren nach einem der Ansprüche 12 bis 14, dadurch gekennzeichnet, dass der Authentifizierungscode (HMAC) getrennt von dem Anwendungsprogramm (Flashware) übermittelt wird.

17. Verfahren nach Anspruch 16, dadurch gekennzeichnet, dass das Anwendungsprogramm (Flashware) auf einem Speichermedium zwischengespeichert und mittels des Speichermediums vertrieben wird und der Authentifizierungscode (HMAC) mittels Datenübertragung vom gesicherten Bereich (Trust-Center) an die Systemschnittstelle übertragen wird.

18. Verfahren nach Anspruch 15, dadurch gekennzeichnet, dass das Anwendungsprogramm (Flashware) und der Authentifizierungscode (HMAC) mittels Datenübertragung vom gesicherten Bereich (Trust-Center) an die Systemschnittstelle übertragen werden.

19. Verfahren nach einem der Ansprüche 12 bis 18, dadurch gekennzeichnet, dass der Authentifizierungscode über die Diagnoseschnittstelle (Diagnose Interface) in ein Steuergerät (ECU) eines Kraftfahrzeuges eingelesen wird.

20. Verfahren nach einem der Ansprüche 12 bis 19, dadurch gekennzeichnet, dass wenn ein eingelesener Authentifizierungscode (HMAC) und im Steuergerät berechneter Hash-Wert übereinstimmen, das zugehörige Anwendungsprogramm (Flashware) mit einer Kennung (Flag) als gültiges Anwendungsprogramm versehen wird.

21. Verfahren nach einem der Ansprüche 12 bis 20, dadurch gekennzeichnet, dass in den Authentifizierungscode (HMAC) Flashware-Metainformation mit einbezogen wird.

22. Verfahren nach Anspruch 21, dadurch gekennzeichnet, dass mit dem Authentifizierungscode (HMAC) der Downloadprozess des Anwendungsprogramms auf verschiedene Steuergeräte selektiert wird.

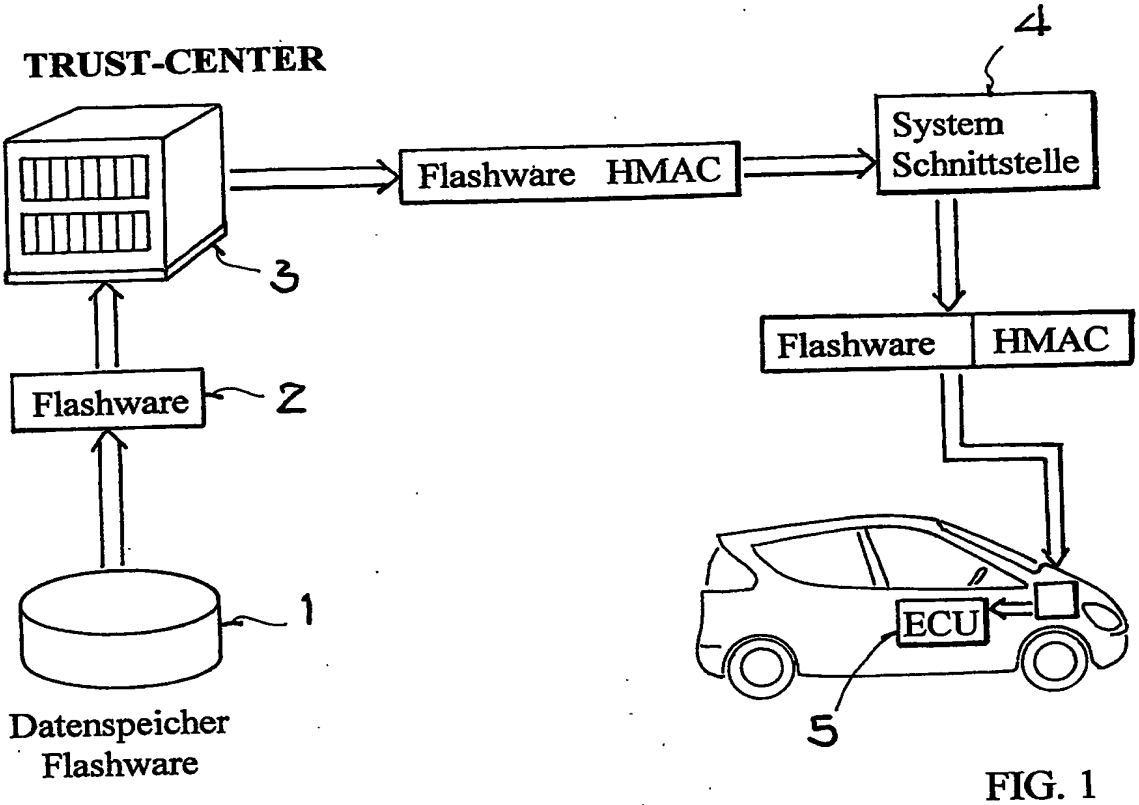
TRUST-CENTER
4
System Schnittstelle
Flashware HMAC
3
Flashware HMAC
Flashware
2
1
ECU
5
Datenspeicher Flashware

FIG. 1

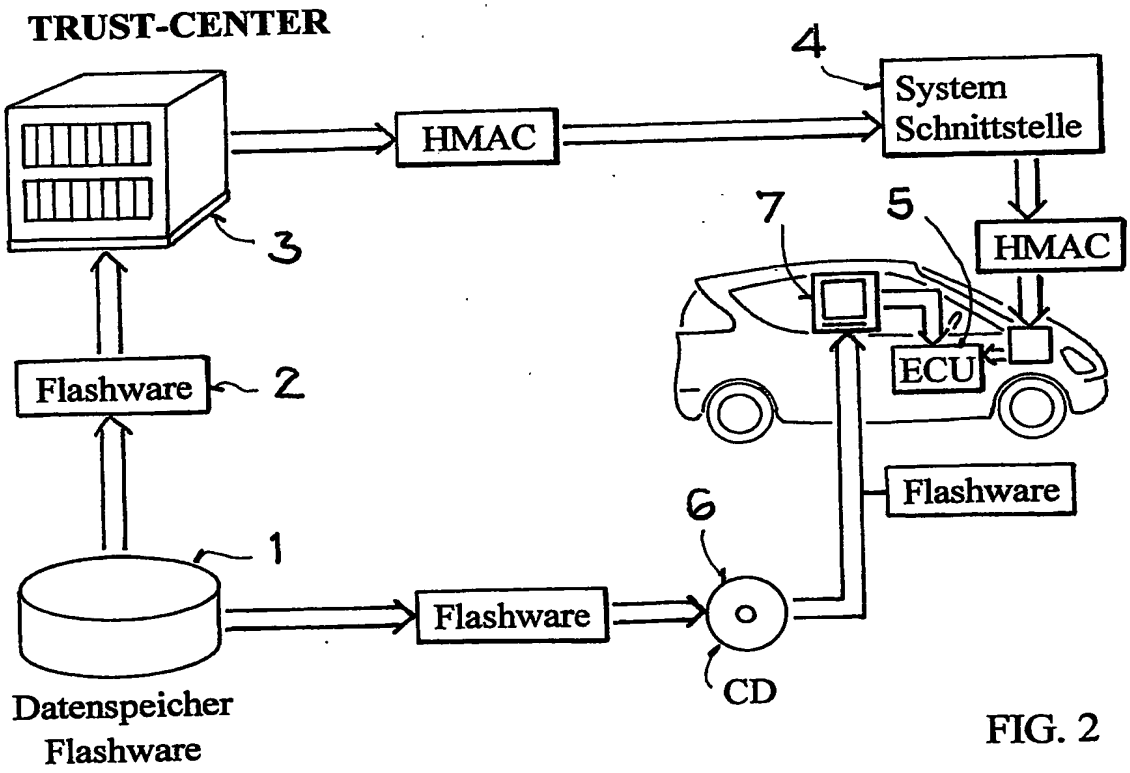
TRUST-CENTER
4
System Schnittstelle
HMAC
3
7
5
HMAC
ECU
Flashware
2
6
1
Flashware
CD
Datenspeicher Flashware

FIG. 2

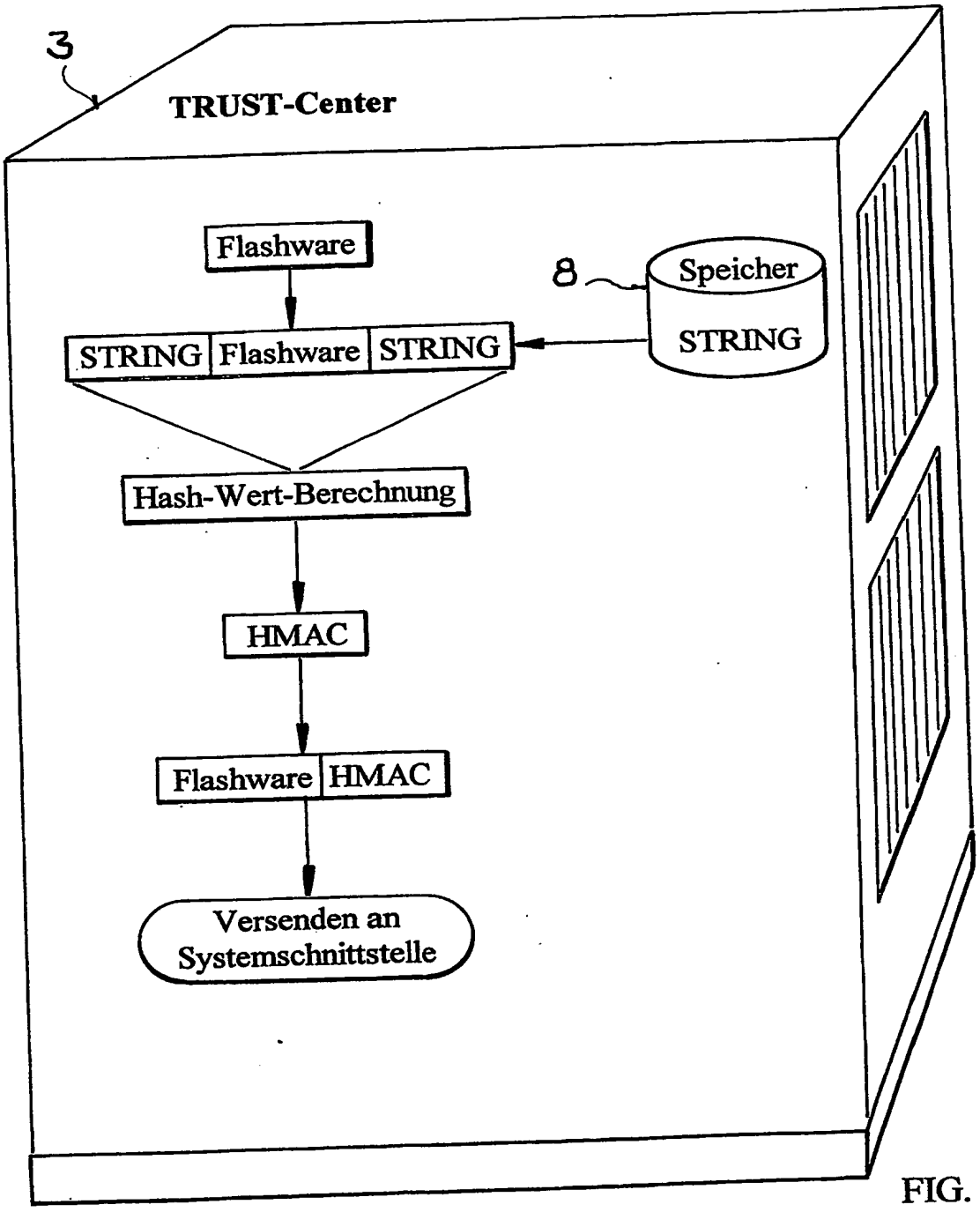
3
TRUST-Center
Flashware
8
Speicher
STRING
STRING
Flashware
STRING
Hash-Wert-Berechnung
HMAC
Flashware
HMAC
Versenden an
Systemschnittstelle

FIG. 3

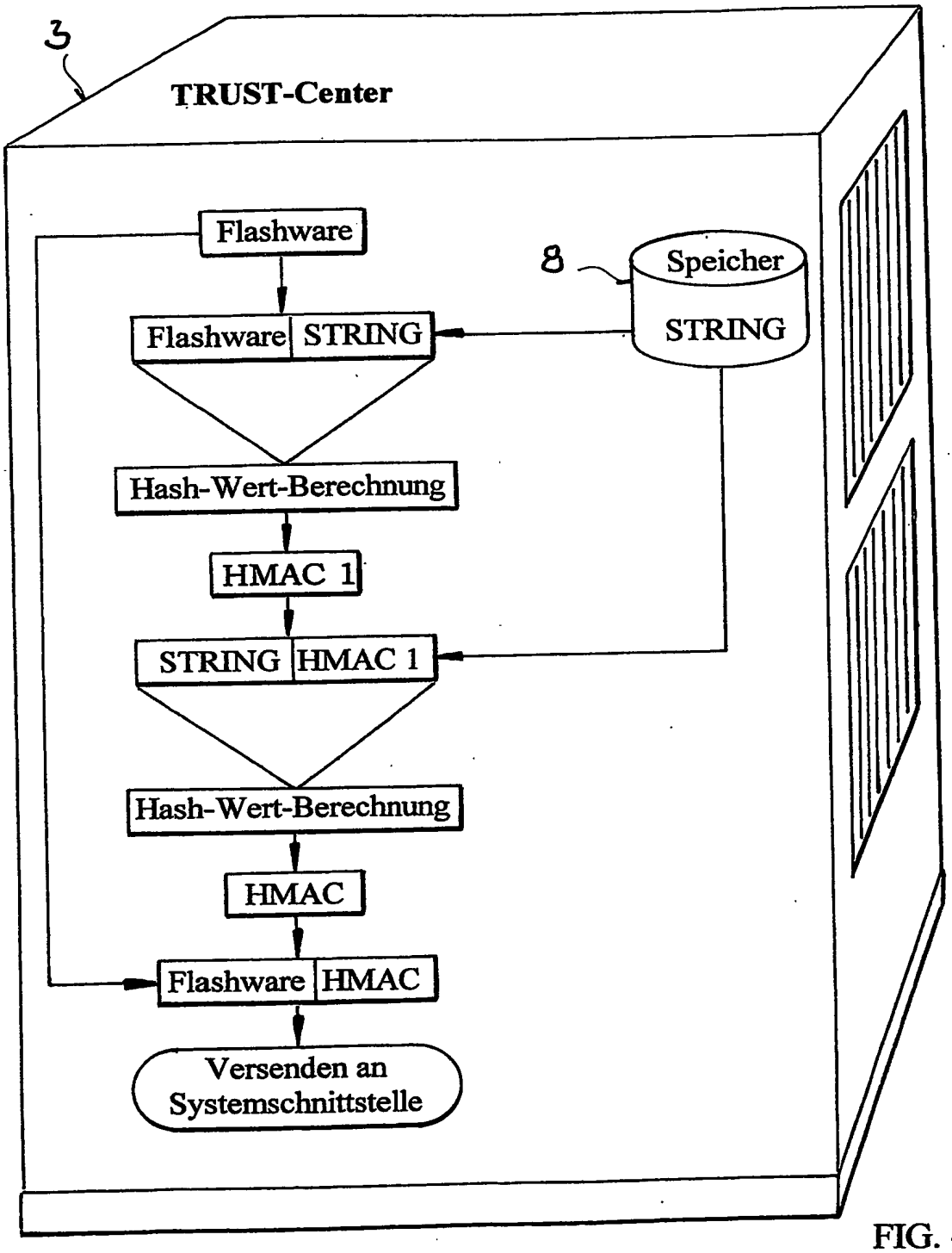
3
TRUST-Center
Flashware
8
Speicher
Flashware
STRING
STRING
Hash-Wert-Berechnung
HMAC 1
STRING
HMAC 1
Hash-Wert-Berechnung
HMAC
Flashware
HMAC
Versenden an
Systemschnittstelle

FIG. 4

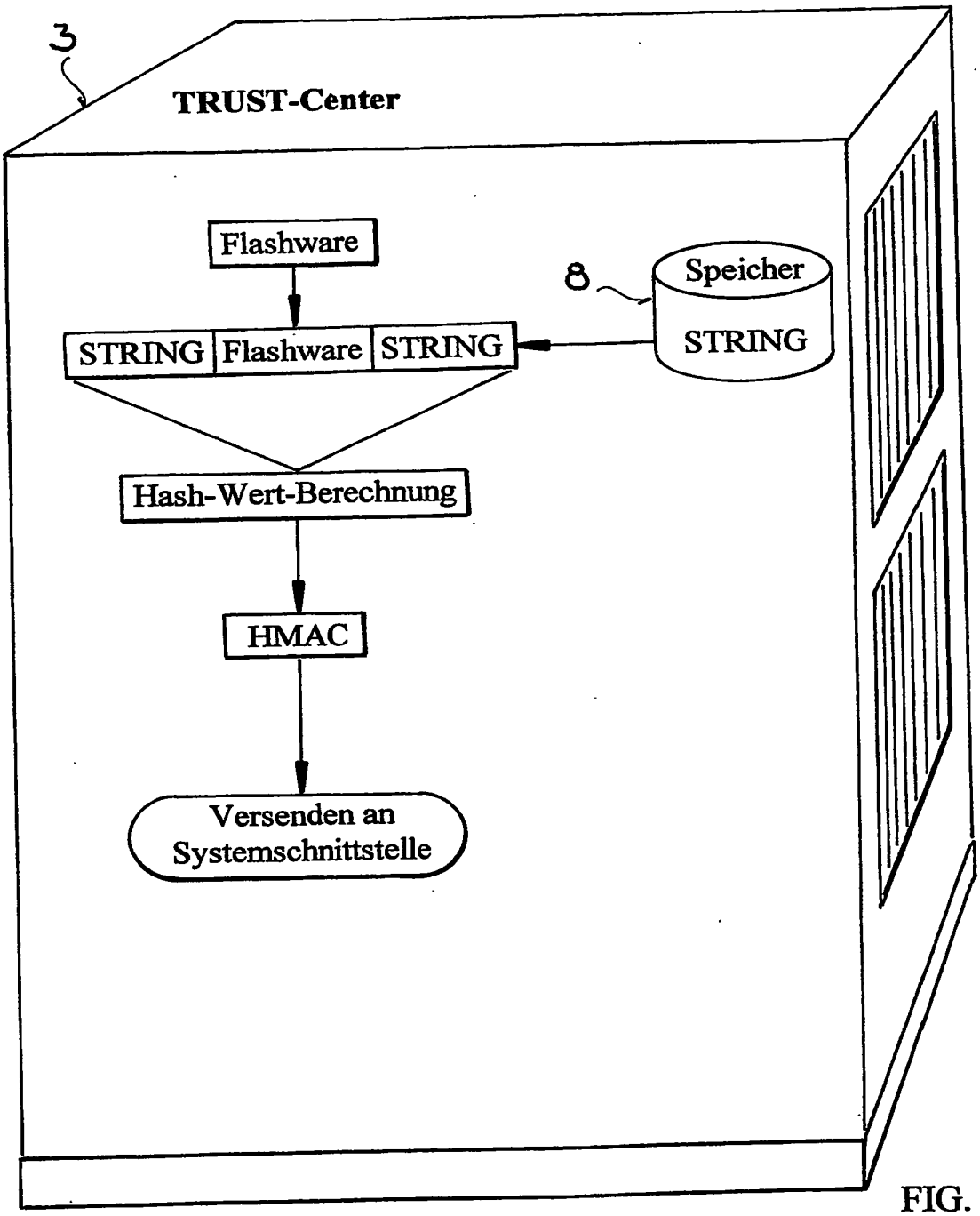
3
TRUST-Center
Flashware
8
Speicher
STRING
STRING
Flashware
STRING
Hash-Wert-Berechnung
HMAC
Versenden an
Systemschnittstelle

FIG. 5

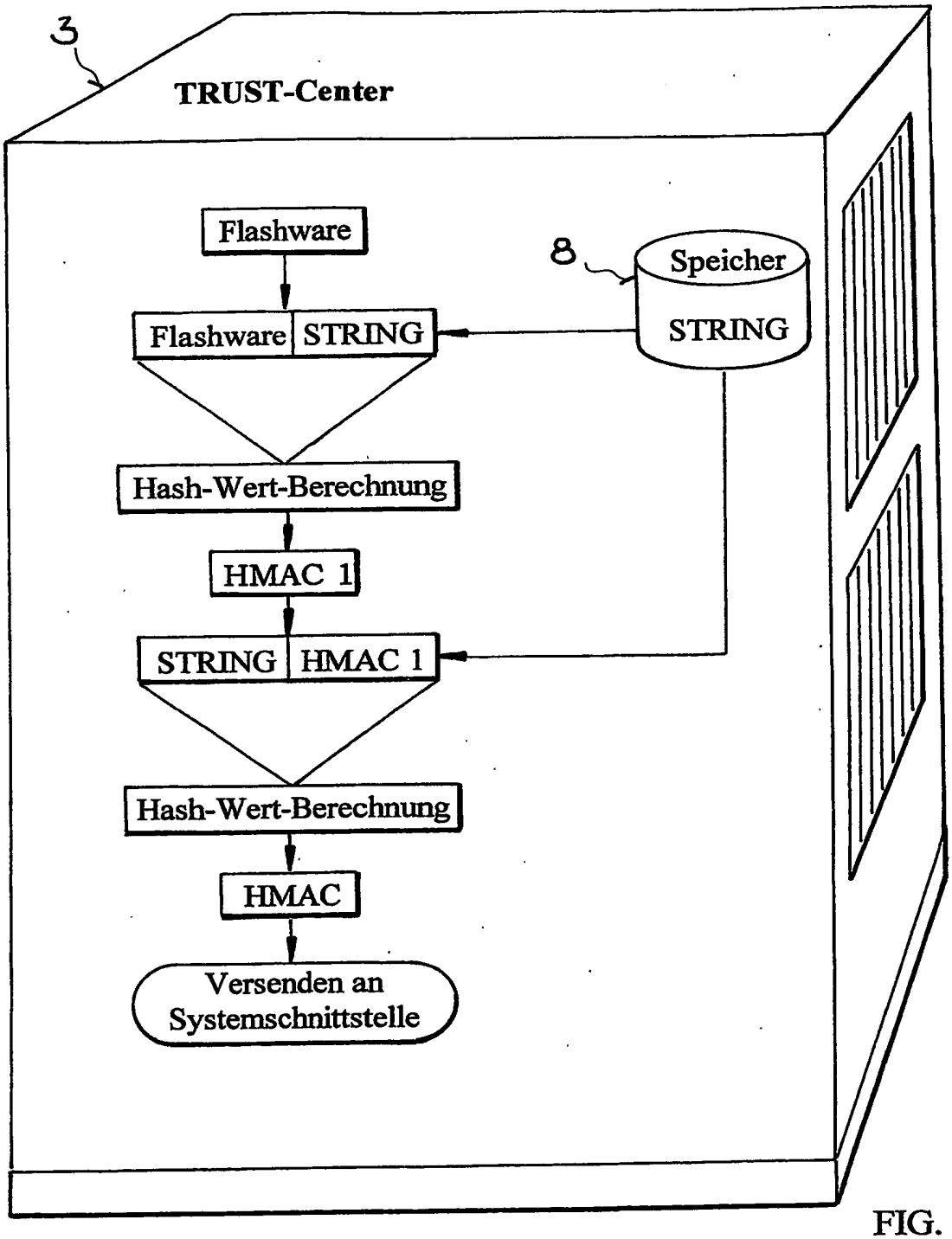
3
TRUST-Center
Flashware
8
Speicher
Flashware
STRING
STRING
Hash-Wert-Berechnung
HMAC 1
STRING
HMAC 1
Hash-Wert-Berechnung
HMAC
Versenden an
Systemschnittstelle

FIG. 6

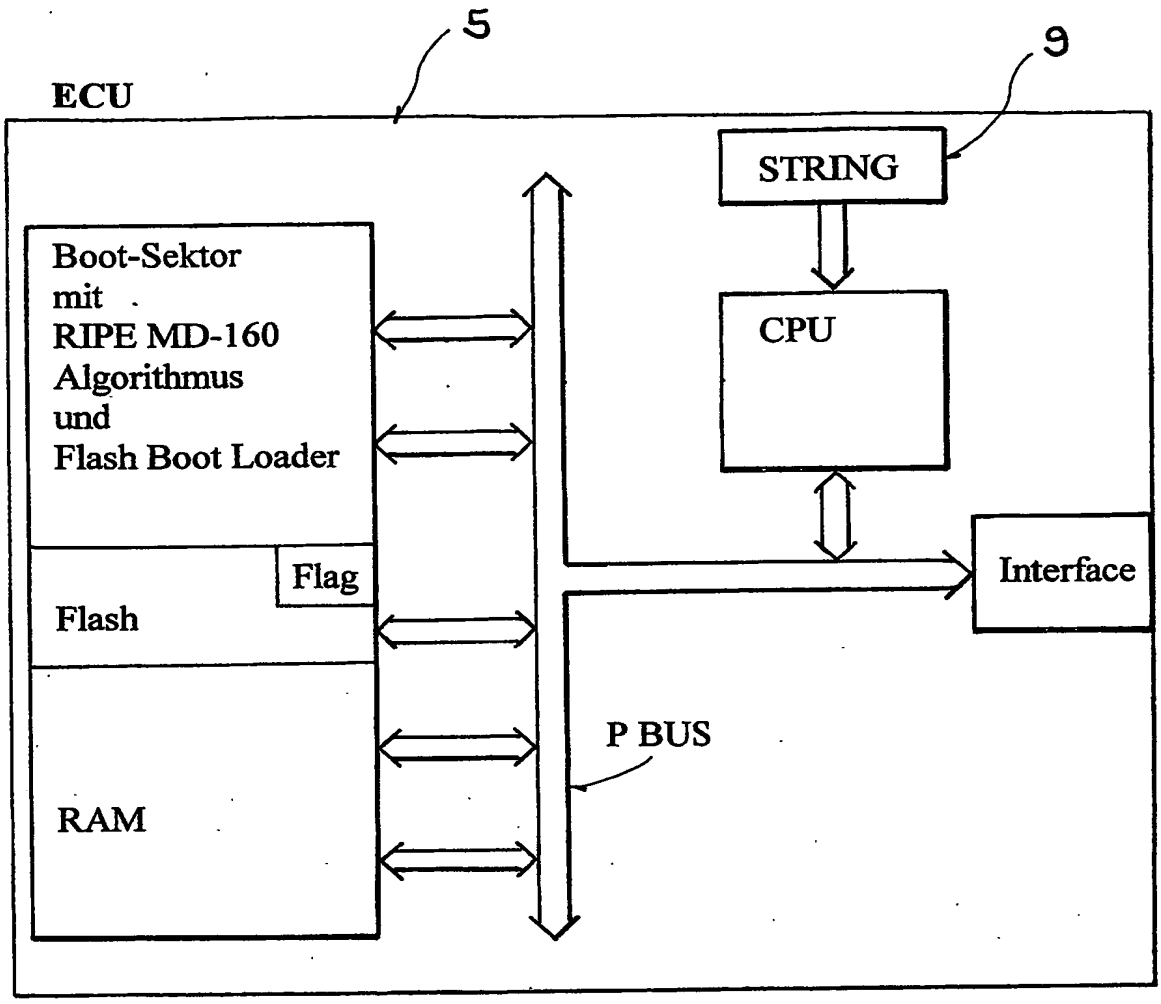
5
9
ECU
STRING
CPU
Boot-Sektor
mit
RIPE MD-160
Algorithmus
und
Flash Boot Loader
Flag
Flash
RAM
Interface
P BUS

FIG. 7

DaimlerChrysler AG

Eschbach
14.04.2003

Zusammenfassung

Die Erfindung betrifft ein vereinfachtes symmetrisches, kryptographisches Verfahren. Grundlage dieses Verfahrens ist ein Authentifizierungscode. Dieser Authentifizierungscode wird in einem gesicherten Bereich, einem sogenannten Trust-Center, berechnet, indem das Anwendungsprogramm, die sogenannte Flashware, mit einem geheimen Datenstring konkateniert wird und von dem konkatenierten Anwendungsprogramm ein Hash-Wert berechnet wird. Dieser Hash-Wert wird hierbei sowohl über das Anwendungsprogramm als auch über den geheimen Datenstring berechnet. Dieser Hash-Wert ist der Authentifizierungscode für das zu prüfende Anwendungsprogramm. Die Überprüfung des Authentifizierungscodes erfolgt in dem Mikroprozessorsystem oder in dem Steuergerät, in dem das Anwendungsprogramm eingesetzt werden soll. Hierzu ist in dem Mikroprozessorsystem oder dem Steuergerät ein zweiter, gleicher, geheimer Datenstring abgelegt. In das Mikroprozessorsystem bzw. in das Steuergerät wird zunächst das unverschlüsselte Anwendungsprogramm und der Authentifizierungscode übertragen. Dann wird im Mikroprozessorsystem bzw. im Steuergerät das unverschlüsselte Anwendungsprogramm mit dem zweiten gleichen, geheimen Datenstring konkateniert. Von diesem konkatenierten Anwendungsprogramm wird im Mikroprozessorsystem bzw. im Steuergerät ein Hash-Wert berechnet. Stimmen berechneter Hash-Wert und übertragener Authentifizierungscode überein, so gilt das übertragene Anwendungsprogramm bzw. die übertragene Flashware als authentisch und darf im Flashspeicher abgelegt werden und im Steuergerät bzw. im Mikroprozessorsystem ange-

wandt werden. In einer Weiterbildung der Erfindung wird das Anwendungsprogramm mit dem geheimen Datenstring beidseitig sowohl am Programmanfang als auch am Programmende konkateniert. Die Hash-Wertberechnung erfolgt dann über das beidseitig konkatenierte Anwendungsprogramm. Zur Überprüfung des dermaßen gebildeten Authentifizierungscodes wird im Mikroprozessorsystem bzw. im Steuergerät das unverschlüsselt übertragene Anwendungsprogramm mit dem im Steuergerät abgelegten zweiten, geheimen Datenstring ebenfalls beidseitig konkateniert und über das beidseitig konkatenierte Anwendungsprogramm im Steuergerät bzw. im Mikroprozessorsystem ein Hash-Wert gebildet. Stimmt der im Steuergerät bzw. Mikroprozessorsystem berechnete Hash-Wert mit dem übertragenen Authentifizierungscode überein, so gilt das übertragene Anwendungsprogramm als authentisch. Die beidseitige Konkatenierung hat den Vorteil eines verbesserten Schutzes gegenüber unerlaubten Manipulationen der Anwendungssoftware.